REMARKS

Claims 1-27 are pending. All stand rejected. The applicants hereby request further consideration and re-examination in view of the remarks set forth below.

Rejections under 35 U.S.C. § 102:

Claims 1 and 25 were rejected under 35 U.S.C. § 102 as being anticipated by Karlsson et al., "Do We Need Replica Placement Algorithms in Content Delivery Networks?" (hereinafter, "Karlsson et al.").

The applicants respectfully traverse the rejection. Claim 1 claims a method of selecting a heuristic class for data placement in a distributed storage system. An integer program is formed for each of a plurality of heuristic classes. Each of the integer programs comprises an objective of minimizing a replication cost. Each of the integer programs is solved, which provides the replication cost for each of the heuristic classes. The heuristic class having a low replication cost is selected.

Regarding claim 1, the office action alleges that Karlsson et al. teaches forming an integer program for each of a plurality of heuristic classes and solving each of the integer programs by teaching: "algorithm that modifies cost function, problem definition specifying cost function; table 2 lists heuristics, techniques for placing data objects in storage nodes" and "achieve goal is cost function simplification and minimization, thereby selection of lower cost replication...Section 2.2, Replica Placement Algorithms."

The applicants disagree with this reasoning. In order anticipate a claim, the prior art reference being relied upon must teach each and every element of the claim. Manual of Patent Examining Procedure, at Section 2131 (8^{th} Ed., Aug. 2006).

Karlsson et al. is missing several of the elements of applicants' claims 1 and 25. Particularly, in Section 2.2, Karlsson et al. discuss that replica placement algorithms consist of a problem definition and a heuristic. Karlsson et al. also discuss that the problem definition consists of a cost function that has to be minimized or maximized under some constraints. Karlsson et al. further explain that heuristics used to achieve the goal set out by the problem definition can be described using three primitives: metric scope, approximation method and cost function simplification. Table 2 lists heuristics examined by Karlsson et al. However, nowhere do Karlsson et al. teach or suggest

forming or solving an integer program. Therefore, Karlsson et al. also do not teach or suggest forming an integer program for each of a plurality of heuristic classes, nor solving each of the plurality integer programs to provide the replication cost for each of the heuristic classes. Karlsson et al. also do not teach selecting a heuristic class having a low replication cost as determined by solving the integer programs. Yet, all of these limitations are recited by the applicants' claims 1 and 25. For at least these reasons, claims 1 and 25 are allowable over Karlsson et al.

Rejections under 35 U.S.C. § 103:

Claims 1-27 were rejected under 35 U.S.C. § 103 as being unpatentable over U.S. Patent No. 6,374,227 to Ye (hereinafter "Ye") in view of Karlsson et al.

Regarding claims 1, 2, 24, 25, 26 and 27, the office action alleges that Ye teaches:

- of [sic] selecting a heuristic class for data placement in a distributed storage system comprising the steps of: (Abstract, line 1 and 12-14 – State that this optimizes allocation of a resource through the specification of a first heuristic)
- forming an integer program for each of a plurality of heuristic classes (Abstract, lines 4-6 – State that an integer program is received for each of a plurality of bids)
- each of the integer programs comprising an objective of minimizing a replication cost (Abstract, lines 14-17 – State that the integer program optimizes allocation, which means that replication cost must therefore be minimized)
- solving each of the integer programs (Abstract, lines 6-7 – State that a solution is generated)
- which provide the replication cost for each of the heuristic classes; (Abstract, lines 17-22 – State that there is a maximization problem)
- selecting the heuristic class having a low replication cost (Abstract, lines 7-10 – State that there is an optimizer engine that is coupled to the file and solver, so in essence, the heuristic class with a low replication cost is selected)
- solving the specific integer program which provides a specific lower bound for the replication cost; (Column 9, lines 44-49 – Disclose a theoretical lower bound which more closely approximates the optimal solution)
- [s]electing the heuristic class if a difference between the general lower bound and the specific lower bottom is within an allowable amount. (Column 15, lines 50-64 – State that the first heuristic is applied if there is a valid value between the upper and lower cutoff.

The office action further states that Ye does not teach "each heuristic class providing a technique for placing data within a distributed system." However, the office action alleges that Karlsson et al. discloses "heuristic classes providing a technique for placing data with the distributed storage system, each of the integer programs comprising an objective of minimizing a replication cost for placing the data." The office action alleges that it would have been obvious to one having ordinary skill in the art "to combine the two systems by providing a technique for placing data within the distributed storage system, as previously known and taught by Karlsson because the result would have provided improved replica algorithm, problem definition simplifying cost function in content delivery networks."

The applicants respectfully traverse the rejection for several reasons. First, the office does not make out a *prima facie* case of obviousness. Second, the Ye and Karlsson et al. references have not been properly combined. Third, even if the Ye and Karlsson et al. references could be combined, they do not teach or suggest each limitation of the rejected claims.

To establish a *prima facie* case of obviousness, three basic criteria must be met: first, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings; second, there must be a reasonable expectation of success; and finally, the prior art reference references must teach or suggest all the claim limitations. MPEP at Section 2143 (Aug. 2006). The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in applicant's disclosure. *Id.* at Section 2143, citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991). Moreover, when applying 35 U.S.C. § 103, the following tenets of patent law must be adhered to: (A) the claimed invention must be considered as a whole; (B) the references must be considered as a whole and must suggest the desirability and thus the obviousness of making the combination; (C) the references must be viewed without the benefit of impermissible hindsight vision afforded by the claimed invention; and (D) reasonable expectation of success is the standard with which obviousness is determined. MPEP at Section 2141

(Aug. 2006) (citations omitted). When these principles are followed, it is apparent that the applicants' claimed invention is not obvious in view of the cited references.

First, the applicants respectfully submit that the office action does not make out a *prima facie* case of obviousness. This is at least because each of the rejected claims has not been considered as a whole. This is clear because the office action does not separately discuss each of the rejected claims, but instead, combines limitations from applicants' claims 1, 2, 25 and 26 into a single bullet list. Claims 24 and 27 do not appear to have been considered at all since their specific limitations are not discussed in the office action. The bullet list of claim limitations shows that rather than considering each claim separately, the rejection is based on hindsight reconstruction in which the claim limitations are alleged to be found in the prior art without regard to the context or meaning of the limitations within the claim as a whole. For at least this reason, the applicants respectfully request that the rejection of claims 1-27 be removed.

Second, the applicants respectfully submit that the Ye and Karlsson et al. references have not been properly combined. This is because there would not have been a suggestion to combine the references in the manner suggested in the office action.

Karlsson et al. quantitatively study replica placement algorithms (RPAs) in content delivery networks (CDNs) in order to discover under what system conditions and performance metrics, if any, RPAs outperform caching. Karlsson et al. evaluate the performance of RPAs by simulating the proposed placements' impact on the system's client-perceived latency. Based on their experimental results, Karlsson et al. conclude that RPAs are not needed for the systems they considered, as a simple caching algorithm can be made to perform as well, if not better than, the best RPAs. However, they believe that RPAs will become useful once availability, reliability and update-propagation latency guarantees have to be provided in a CDN. See, Karlsson et al. at Section 1, "Introduction."

In contrast, Ye is directed toward a system and method for optimizing allocation of a resource. Ye at col. 1, lines 11-13. Ye explains that businesses and other organizations wish to optimize the manner in which they allocate various resources, to reduce costs and improve efficiency. Ye at col. 1, lines 16-18. For example, a shipper of goods may wish to optimize the allocation of transportation capacity among multiple

competing carriers in a way that minimizes shipping costs while satisfying the shipper's coverage, equipment, service and other requirements. Ye at col. 1, lines 18-23. Ye further explains that conventional integer program solvers, such as CPLEX and XPRESS may use a standard branch and bound approach in an attempt to solve optimization problems that are formulated as integer programs, but are often unable to solve even relatively small such problems in a brute force manner before overloading the memory of even the most powerful computers. Ye at col. 1, lines 37-43. Ye proposes a system for optimizing the allocation of a resource that includes an optimizer file containing resource allocation data including a demand for allocation of the resource, a plurality of bids for the resource, and a plurality of reserve bids for the resource. Ye at col. 1, lines 57-62. A solver receives an integer program and generates an LP relaxation solution to the integer program. Ye at col. 1, lines 62-63. An optimizer engine coupled to the file and to the solver receives the data and the LP relaxation solution and generates an "enhanced" integer program. Ye at col. 1, line 63 to col. 2, line 3. The solver generates a solution to the enhanced integer program that optimizes the allocation of the resource subject to the demand, bids, and reserve bids. Ye at col. 2, lines 4-6.

Accordingly, it is apparent that Karlsson et al. and Ye, when considered as a whole - as they must, are directed to entirely different fields of endeavor. Particularly, Karlsson et al. study replica placement algorithms (RPAs) in content delivery networks (CDNs) in order to discover under what system conditions and performance metrics, if any, RPAs outperform caching. In contrast, Ye is directed toward a system and method for optimizing allocation of a resource by generating a solution to an "enhanced" integer program. Because of their disparate teachings, there would not have been a motivation to combine them.

As best understood by the applicants, the office action asserts that a person would have been motivated to combine Karlsson et al. and Ye because the result would have provided an improved RPA. The applicants respectfully disagree. While Karlsson et al. discuss RPAs in content delivery networks, Karlsson et al. do so in the context of comparing RPAs to caching algorithms. See, Karlsson et al. at Section 4.3 "Comparison against caching." Karlsson et al. do not attempt to provide an improved RPA. Morever, there is no suggestion that the combination of Karlsson et al. with Ye would result in an

improved RPA. There is also no suggestion of how Ye might be applied to the teachings of Karlsson et al. to result in an improved RPA. For at least these reasons, the applicants respectfully request that the rejection of claims 1-27 be removed.

Third, even if they Ye and Karlsson et al. references could be combined, they do not teach or suggest each limitation of the rejected claims. For example, applicant's claim 1 is directed toward a method of selecting a heuristic class for data placement in a distributed storage system by forming an integer program for each of a plurality of heuristic classes, solving each of the integer programs, which provides the replication cost for each of the heuristic classes and selecting the heuristic class having a low replication cost. Karlsson et al. do not teach or suggest forming or solving an integer program. Therefore, Karlsson et al. also do not teach or suggest forming an integer program for each of a plurality of heuristic classes, nor solving each of the plurality integer programs to provide the replication cost for each of the heuristic classes. Karlsson et al. also do not teach selecting a heuristic class having a low replication cost as determined by solving the integer programs. While Ye discusses integer programs, Ye does not teach or suggest forming an integer program for each of a plurality of heuristic classes, nor solving each of the plurality integer programs to provide the replication cost for each of the heuristic classes, nor does Ye teach selecting a heuristic class having a low replication cost as determined by solving integer programs. For at least these reasons, claim 1 is allowable over Karlsson et al. and Ye, taken singly or in combination. Claim 25 recites a computer readable memory comprising computer code for implementing the method of claim 1. Accordingly, claim 25 is allowable for at least the same reasons that claim 1 is allowable.

Regarding claim 1, the office action asserts that Ye teaches forming an integer program for each of a plurality of heuristic classes because Ye teaches that "an integer program is received for each of a plurality of bids." The applicants respectfully disagree with this reasoning. First, it mis-characterizes Ye at least because Ye instead teaches that the bids are inputs to the process of Ye; nowhere does Ye state that an integer program is formed for each bid. Second, even if Ye did teach forming an integer program for each of a plurality of bids, this still would not anticipate forming an integer program for each of a plurality of heuristic classes. This is at least because the bids of Ye do not anticipate,

and are unrelated to, the heuristic classes of applicant's claim 1. Rather, the bids of Ye are simply offers by carriers to ship goods. See, Ye at col. 4, lines 22-42.

Also regarding claim 1, the office action asserts that Ye teaches selecting the heuristic class having a low replication cost because Ye teaches that "there is an optimizer engine coupled to the file and the solver, so in essence, the heuristic class with a low replication cost is selected." The applicants respectfully disagree with this reasoning. While Ye does discuss cost, Ye states that the optimizer awards bids and reserve bids among multiple competing carriers in a way that minimizes total shipping cost. Ye at col. 4, lines 42-45. However, as explained above, the bids of Ye do not anticipate, and are unrelated to, the heuristic classes of applicant's claim 1. Further, Ye teaches nothing about replication cost for placing data in a distributed storage system.

These are additional reasons why claims 1 and 25 are allowable.

Claim 2 claims a method of selecting a heuristic class for data placement in a distributed storage system. A general integer program, which models the data placement, is formed. A specific integer program, which models a heuristic class for the data placement, is formed. The general and specific integer programs each comprise an objective of minimizing a replication cost. The general integer program is solved, which provides a general lower bound for the replication cost. The specific integer program is solved, which provides a specific lower bound for the replication cost. The heuristic is selected if a difference between the general lower bound and the specific lower bound is within an allowable amount.

Neither Karlsson et al., nor Ye, teaches or suggests the formation and solving of both a general integer program and a specific integer program. Rather, as explained above, Karlsson et al. do not teach or suggest forming or solving any integer programs. Ye also does not teach or suggest this feature. This is because Ye teaches only that an "enhanced" integer program is solved. See, Ye at col. 2, lines 4-6. Moreover, neither Karlsson et al., nor Ye, teaches or suggests solving a general integer program and a specific integer program to determine a general lower bound and a specific lower bound for replication cost. Karlsson et al. cannot teach this feature because Karlsson et al. do not teach or suggest forming or solving any integer program. Ye also does not teach or suggest this feature. While Ye states that LP relaxation provides a lower bound for its

integer program, Ye does not teach general and specific lower bounds for replication cost. See, Ye at col. 5, lines 55-61.

Moreover, neither Karlsson et al., nor Ye, teaches or suggests selecting a heuristic class based on comparison of a general lower bound and a specific lower bound. The office action states that Ye teaches this feature by its teaching that "the first heuristic is applied if there is a valid value between the upper and lower cutoff." The applicants respectfully disagree with this reasoning. First Ye teaches nothing about selecting a heuristic class. Rather, the heuristic referred to by Ye is used to generate the "enhanced" integer program. See, Yee at col. 12, lines 16-23. As such, there is no teaching in Ye of selecting of a heuristic class and there is no teaching in Ye of selecting of a heuristic class based on of a general lower bound and a specific lower bound determined by forming and solving integer programs.

For at least these reasons, claim 2 is allowable over Karlsson et al. and Ye, taken singly or in combination. Claim 26 recites a computer readable memory comprising computer code for implementing the method of claim 2. Accordingly, claim 26 is allowable for at least the same reasons that claim 2 is allowable.

Claims 3-23 are allowable at least because each depends from an allowable base claim 1. In the interest of brevity, the Office Action assertions regarding claims 3-23 are not being individually addressed here. Applicant also notes that many if not all of these assertions are incorrect. For example, with regards to claim 3, the Office Action refers to Ye at col. 6, line 67, to col. 7, line 2, and col. 21, lines 29-41 for the proposition that Ye teaches, "wherein inputs used in the steps of forming the general and specific integer programs comprise a system configuration," which is incorrect. Ye at col. 6, line 67, to col. 7, line 2, refers to a computer used to solve an integer program, not a configuration of a computer or a computer system (i.e., a system configuration) that is an input to an integer program. And, Ye at col. 21, lines 29-41, refers to assigning priorities to bids and ordering bids according to weighted unit price.

Claim 24 claims a method of selecting a heuristic class for data placement in a distributed storage system. A general integer program, which models the data placement, is formed. Specific integer programs, which model a plurality of heuristic classes, are formed. The general and specific integer programs each comprise an objective of

minimizing a replication cost. The general integer program is solved, which provides a lower bound for the replication cost. The specific integer programs are solved, which provides the replication cost for each of the heuristic classes. A particular heuristic class correlated to a low replication cost is selected if a difference between the lower bound and the low replication cost is within an allowable limit.

As explained above with respect to claim 2, neither Karlsson et al., nor Ye, teaches or suggests the formation and solving of both a general integer program and a specific integer program. Karlsson et al. and Ye also do not teach or suggest solving a plurality of specific integer programs. Rather, as is also explained above, Karlsson et al. do not teach or suggest forming or solving any integer programs. Ye also does not teach or suggest this feature. This is because Ye teaches only that an "enhanced" integer program is solved. See, Ye at col. 2, lines 4-6. Moreover, neither Karlsson et al., nor Ye, teaches or suggests solving a general integer program and a plurality of specific integer programs to determine a general lower bound for replication cost and the replication cost for each of a plurality of heuristic classes. Karlsson et al. cannot teach this feature because Karlsson et al. do not teach or suggest forming or solving any integer program. Ye also does not teach or suggest this feature. While Ye states that LP relaxation provides a lower bound for its integer program, Ye does not teach a general lower bound for replication cost, nor a replication cost for each of a plurality of heuristic classes. See, Ye at col. 5, lines 55-61.

Moreover, neither Karlsson et al., nor Ye, teaches or suggests selecting a heuristic class correlated to a low replication cost if a difference between the lower bound and the low replication cost is within an allowable limit. Rather, Ye teaches nothing about selecting a heuristic class. The heuristic referred to by Ye is used to generate the "enhanced" integer program. See, Yee at col. 12, lines 16-23. As such, there is no teaching in Ye of selecting of a heuristic class and there is no teaching in Ye of selecting of a heuristic class based on correlation to a low replication cost if a difference between the lower bound and the low replication cost is within an allowable limit.

For at least these reasons, claim 24 is allowable over Karlsson et al. and Ye, taken singly or in combination. Claim 27 recites a computer readable memory comprising

computer code for implementing the method of claim 24. Accordingly, claim 27 is allowable for at least the same reasons that claim 24 is allowable.

Conclusion:

In view of the above, the Applicant submits that all of the pending claims are now allowable. Allowance at an early date would be greatly appreciated. Should any outstanding issues remain, the Examiner is encouraged to contact the undersigned at (408) 293-9000 so that any such issues can be expeditiously resolved.

Respectfully Submitted,

Dated: March 23, 2007

Derek J. Westberg (Reg. No. 40,872)